

**CHARACTER GENERATING DEVICE AND ITS METHOD**

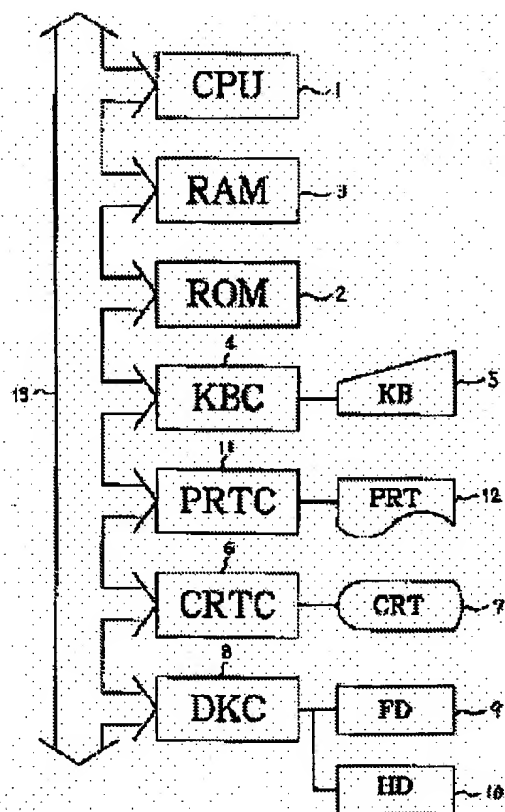
Patent number: JP8016143  
Publication date: 1996-01-19  
Inventor: MOROOKA HIDEKAZU  
Applicant: CANON KK  
Classification:  
- international: G09G5/24; G09G5/24; (IPC1-7): G09G5/24  
- european:  
Application number: JP19940144462 19940627  
Priority number(s): JP19940144462 19940627

Report a data error here

**Abstract of JP8016143**

**PURPOSE:**To reduce the data size of a bit map to be mounted in a device and to hold a sufficient character quality by switching the generation class of a character outline according to the outline size of character data.

**CONSTITUTION:**A CPU 1 functions as an outline generating means generating the outline from plural font files mounted in this system and sets outline generating selection data for switching whether the outline is to be generated from outline fonts or from hybrid fonts in the outline generating selection table on an RAM 3. Then, the CPU 1 reads out previously stored character data with respect to output characters and specific discrimination information and decides the generation class of the character outline with the output size of the character data to perform the switching of the generation class of the character outline based on the decision result and then allows a bit map font pattern to be generated based on the generation class of the character outline after this switching.



Data supplied from the esp@cenet database - Worldwide

**BEST AVAILABLE COPY**

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-16143

(43) 公開日 平成8年(1996)1月19日

(51) Int. Cl. <sup>6</sup>  
G09G 5/24

識別記号

9377-5H

F I

審査請求 未請求 請求項の数12 O L (全10頁)

(21) 出願番号 特願平6-144462

(22) 出願日 平成6年(1994)6月27日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 師岡 秀和

東京都大田区下丸子3丁目30番2号 キヤ

ノン株式会社内

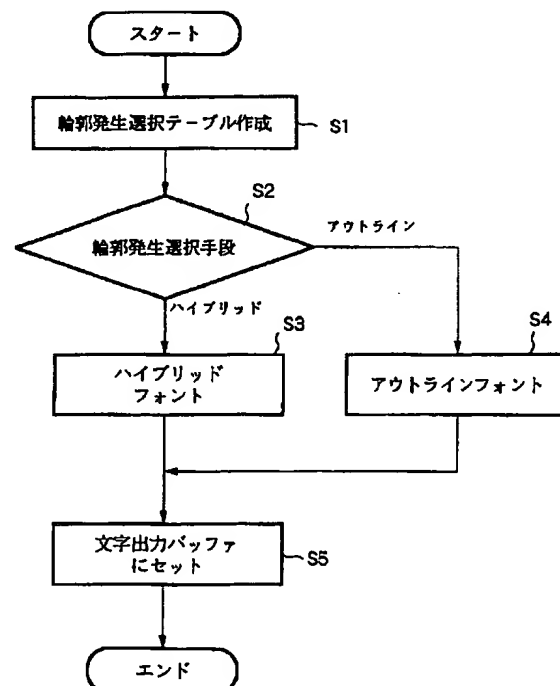
(74) 代理人 弁理士 大塚 康德 (外1名)

(54) 【発明の名称】 文字発生装置及びその方法

(57) 【要約】

【目的】 搭載するビットマップデータサイズを大幅に減らし、十分な文字品位を保つ。

【構成】 システムに搭載されている複数のフォントファイルより、輪郭発生をアウトラインとハイブリッドに切り替えるための輪郭発生選択データを、メモリ上の輪郭発生選択テーブルにセットする。また、入力された書体、文字コード、出力文字サイズ、輪郭発生手段切り替えフラグにより、輪郭発生手段をハイブリッドフォント、またはアウトラインフォントにより生成されたものかを選択する処理を行なう。このように、小さな文字を出力する際に、文字の潰れが発生するエレメントに対してのみビットマップで記憶し、他のエレメントについてはアウトラインにより発生させ、合成する。



## 【特許請求の範囲】

【請求項 1】 複数の文字エレメントをベクトル形式及びビットマップ形式で記憶し、該文字エレメントをエレメント合成情報によって合成して所望の文字パターンを生成する文字発生装置において、

あらかじめ格納された、出力文字についての文字データ及び所定の判定情報を読み出す手段と、

前記判定情報及び前記文字データの出力サイズにより文字輪郭の発生種別を判断する手段と、

前記判断結果に基づいて、前記文字輪郭の発生種別の切り換えを行なう手段と、

前記切り換え後の文字輪郭の発生種別に基づいてビットマップフォントパターンを発生させるビットマップフォントパターン発生手段とを備えることを特徴とする文字発生装置。

【請求項 2】 前記所定の判定情報は前記出力文字のサイズ情報であり、該出力文字の書体単位で格納されていることを特徴とする請求項 1 に記載の文字発生装置。

【請求項 3】 前記ビットマップフォントパターン発生手段は、さらに、前記ベクトル形式に従って前記文字エレメント発生させる手段と、前記文字データの出力サイズに応じた所定の文字エレメントに関して、前記ビットマップ形式にて記憶された該文字エレメントを前記エレメント合成情報により合成して文字を発生する手段とを備えることを特徴とする請求項 1 に記載の文字発生装置。

【請求項 4】 前記所定の文字エレメントは、前記文字データの出力サイズによっては文字品位が低下する文字エレメントであることを特徴とする請求項 3 に記載の文字発生装置。

【請求項 5】 前記ビットマップ形式で記憶されている文字エレメントは、拡大／縮小処理を行なうことによってビットマップエレメントの複数のサイズに適應できることを特徴とする請求項 1 に記載の文字発生装置。

【請求項 6】 前記ビットマップフォントパターン発生手段は、所定の出力サイズを有する文字については、前記ビットマップ形式で記憶されている文字エレメントを取り去ってから、前記エレメント合成情報に従って文字エレメントの合成を行なうことを特徴とする請求項 1 に記載の文字発生装置。

【請求項 7】 複数の文字エレメントをベクトル形式及びビットマップ形式で記憶し、該文字エレメントをエレメント合成情報によって合成して所望の文字パターンを生成する文字発生方法において、

あらかじめ格納された、出力文字についての文字データ及び所定の判定情報を読み出す工程と、

前記判定情報及び前記文字データの出力サイズにより文字輪郭の発生種別を判断する工程と、

前記判断結果に基づいて、前記文字輪郭の発生種別の切り換えを行なう工程と、

前記切り換え後の文字輪郭の発生種別に基づいてビットマップフォントパターンを発生させるビットマップフォントパターン発生工程とを備えることを特徴とする文字発生方法。

【請求項 8】 前記所定の判定情報は前記出力文字のサイズ情報であり、該出力文字の書体単位で格納されていることを特徴とする請求項 7 に記載の文字発生方法。

【請求項 9】 前記ビットマップフォントパターン発生工程は、さらに、前記ベクトル形式に従って前記文字エレメント発生させる工程と、前記文字データの出力サイズに応じた所定の文字エレメントに関して、前記ビットマップ形式にて記憶された該文字エレメントを前記エレメント合成情報により合成して文字を発生する工程とを備えることを特徴とする請求項 7 に記載の文字発生方法。

【請求項 10】 前記所定の文字エレメントは、前記文字データの出力サイズによっては文字品位が低下する文字エレメントであることを特徴とする請求項 9 に記載の文字発生方法。

【請求項 11】 前記ビットマップ形式で記憶されている文字エレメントは、拡大／縮小処理を行なうことによってビットマップエレメントの複数のサイズに適應できることを特徴とする請求項 7 に記載の文字発生方法。

【請求項 12】 前記ビットマップフォントパターン発生工程では、所定の出力サイズを有する文字については、前記ビットマップ形式で記憶されている文字エレメントを取り去ってから、前記エレメント合成情報に従って文字エレメントの合成を行なうことを特徴とする請求項 7 に記載の文字発生方法。

30 【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、文字エレメントをベクトル形式及びビットマップ形式で記憶して文字を発生する文字発生装置に関するものである。

【0002】

40 【従来の技術】 従来より、プリンタ装置等において所望の文字を発生させるため、文字の輪郭をベクトル形式で持ち、あらゆるサイズにスケーリング可能なフォントとしてアウトラインフォントが知られている。しかし、アウトラインフォントには、例えば、60 ドット以下の小さな文字を出力させようとする場合、演算誤差等により文字の一部が潰れ、文字品位が落ちる、という問題がある。

【0003】 そこで、この潰れを回避するために、従来は、小さな文字に対してはビットマップフォントをプリンタ装置等に搭載して、印字出力をしている。

【0004】

50 【発明が解決しようとする課題】 しかしながら、上記従来の方法では、小さな文字に対してビットマップフォントをプリンタ装置等に搭載して印字しても、そのデータ

サイズが膨大になるという問題がある。本発明は、上記の課題に鑑みてなされたもので、その目的とするところは、装置に搭載するビットマップデータサイズを大幅に減らし、十分な文字品位を保てる文字発生装置を提供することである。

【0005】さらに、本発明の目的は、文字を簡略して表示して出力の高速化を行なうことである。

【0006】

【課題を解決するための手段】及び

【作用】上記の目的を達成するため、請求項1に記載の発明は、複数の文字エレメントをベクトル形式及びビットマップ形式で記憶し、該文字エレメントをエレメント合成情報によって合成して所望の文字パターンを生成する文字発生装置において、あらかじめ格納された、出力文字についての文字データ及び所定の判定情報を読み出す手段と、前記判定情報及び前記文字データの出力サイズにより文字輪郭の発生種別を判断する手段と、前記判断結果に基づいて、前記文字輪郭の発生種別の切り換えを行なう手段と、前記切り換え後の文字輪郭の発生種別に基づいてビットマップフォントパターンを発生させるビットマップフォントパターン発生手段とを備える。

【0007】また、請求項3に記載の発明では、前記ビットマップフォント発生手段は、さらに、前記ベクトル形式に従って前記文字エレメント発生させる手段と、前記文字データの出力サイズに応じた所定の文字エレメントに関して、前記ビットマップ形式にて記憶された該文字エレメントを前記エレメント合成情報により合成して文字を発生する手段とを備える。

【0008】上記の構成をとることで、当該装置に搭載するビットマップデータサイズを大幅に減らし、かつ、十分な文字品位を保つよう機能する。また、請求項6に記載の発明は、前記ビットマップフォントパターン発生手段は、所定の出力サイズを有する文字については、前記ビットマップ形式で記憶されている文字エレメントを取り去ってから、前記エレメント合成情報に従って文字エレメントの合成を行なう。

【0009】この構成により、文字出力の高速化が可能となる。

【0010】

【実施例】以下、添付図面を参照して、本発明に係る好適な実施例を詳細に説明する。

【第1実施例】図1は、本発明の第1の実施例に係る、例えば、日本語ワードプロセッサ、ワークステーション、あるいはコンピュータ等の文字発生システムの全体構成を示すブロック図である。同図において、符号1はCPU（中央処理装置）であり、本システム全体の制御及び演算処理等を行なう。ROM（読み出し専用メモリ）2は、システム起動プログラム及びフォントファイル等の記憶領域として使用される。また、RAM（ランダムアクセスメモリ）3は、使用制限のないデータ記憶

領域であり、様々な処理毎に各々のプログラム及びデータがロードされ、実行される領域である。

【0011】KBC（キーボード制御部）4は、KB（キーボード）5からのキー入力データ（文字コード、制御コマンド等）を受け取り、それをシステムバス13を介してCPU1へ伝達する。CRTC（ディスプレイ制御部）6は、CRT（ディスプレイ装置）7を制御し、CRT7は、CRTC6からデータを受け取り、表示する。

【0012】また、DKC（ディスク制御部）8には、外部記憶装置としてのFD（フロッピーディスク装置）9やHD（ハードディスク装置）10が接続され、データ伝送等の制御を行なう。本システムでは、これらの装置にプログラム及びデータを記憶させておき、実行時に必要に応じて、それを参照、またはRAM3へロードする。そして、PRTC（プリンタ制御部）11は、それに接続されるPRT（プリンタ装置）12を制御する。

【0013】なお、システムバス13は、上述の構成要素間のデータの通路となる。以下、上記の構成をとる本実施例に係るシステムの動作について詳細に説明する。最初に、本実施例におけるアウトラインフォント及びハイブリッドフォントについて説明する。

【0014】本実施例に係るシステムにて用いるアウトラインフォントとは、ある文字について、その文字の骨格となる横線、縦線といった基本文字構成要素と、その始点部、終点部の局部輪郭（偏、旁等）を表わす構成要素やその他の文字基本構成要素を修飾する構成要素といった複数の文字構成要素（以後、これらをエレメントという）に分割して記憶し、必要に応じて、各エレメントを組み合わせることで文字を生成する、エレメント合成型のアウトラインフォントである。

【0015】例えば、図2に示す文字において、実線部が基本エレメントであり、破線部が基本エレメントを修飾するエレメントである。そして、これらのエレメントの形状を規定するデータは、図3に示すように、エレメントの芯線を直線により表わし、輪郭をそこから距離で表現する「芯線+太さ」型であっても、図4に示すように、輪郭をベクター形式で表わす「ベクター輪郭型」であっても良い。

【0016】また、各エレメントには、エレメントIDが割り振られており、各文字コードに1対1に対応したエレメント組み合わせ情報には、その文字を合成するためのエレメントID及びエレメント配置情報が記憶されており、それらを用いてエレメントの合成を行なう。一方、ハイブリッドフォントとは、従来、ビットマップフォントを用いて出力していた、例えば、60ドット以下の小さな文字に対して、アウトラインフォントでは潰れ等により文字品位が保てない局部輪郭エレメント（偏、旁等、例えば、図2における破線部）をビットマップ形式で記憶し、それ以外のエレメントに対しては、アウト

ラインフォントよりビットマップを生成し、合成することで、文字を生成するものである。

【0017】図5は、本実施例に係るシステムにおける基本動作手順を示すフローチャートである。同図のステップS1では、CPU1が、本システムに搭載されている複数のフォントファイルより輪郭を発生する輪郭発生手段として機能し、輪郭をアウトラインフォントより発生するか、あるいは、ハイブリッドフォントにより発生するかを切り替えるための輪郭発生選択データ（以後、これを選択データという）を、RAM3上の輪郭発生選択テーブル（以後、選択テーブルという）にセットする。

【0018】図6は、上記の輪郭発生選択テーブルの作成過程を表すフローチャートである。同図のステップS11では、フォントファイルの読み出し終了を判断し、本システム内の全フォントファイルを読み出しが終了した場合、本処理を終えて、図5のステップS2に戻る。しかし、それ以外の場合には、ステップS12へ進み、そこで、選択データをフォントファイルより読み出す。

【0019】ここで扱うフォントファイルとは、図7に示すように書体単位に扱われ、基本的には、「フォントデータ+選択データ」の構造をとるが、ハイブリッドフォントを用いない場合は、選択データが存在しない。ステップS13では、選択データがフォントファイル中に存在するか否かを判定する。ここでの判定の結果、選択データが存在する場合、処理をステップS14に進め、また、選択データが存在しない場合には、処理をステップS11に戻す。

【0020】なお、ステップS14では、選択データをRAM3上の選択テーブルにシーケンシャルに反映させる。図8は、本実施例に係る、ROM2に記憶される輪郭発生選択データの構造を示す図である。ここでは、データの要素として、書体名、ハイブリッドフォントを採用する際の閾値（以後、ハイブリッド閾値という）、ハイブリッドフォントを構成する際に使用するビットマップ数及びビットマップサイズ、ビットマップサイズ間を補完する際に用いるビットマップ拡大、縮小率を記述したテーブル（以後、レイトテーブルという）を持つ。

【0021】図9は、上記レイトテーブルの構造を示す図である。同図に示すように、本レイトテーブルは、要素として、出力ビットマップサイズ、拡大・縮小率、使用ビットマップサイズを持ち、出力サイズが決まると、このテーブルを検索することにより、拡大・縮小率、使用ビットマップサイズが一意に求まる。例えば、58ドットのビットマップを出力したい場合、このレイトテーブルより、出力サイズ58ドットは、拡大・縮小率（レイト）が24/25となり、使用ビットマップサイズは60ドットとなる。ここで、レイトが24/25とは、25ドット目を削除して24ドットとすることを表わし、

この例では、25、50ドット目を削除することにより2ドット減らして58ドットにすることで、58ドットのビットマップを得る。

【0022】図5のステップS2では、入力された書体、文字コード、出力文字サイズ、輪郭発生手段切り替えフラグ（以後、切り替えフラグという）により、輪郭発生手段をハイブリッドフォント、またはアウトラインフォントにより生成されたものかを選択する、輪郭発生選択手段（CPU1に相当）による処理を行なう。図10は、上記の輪郭発生選択の過程を示すフローチャートである。同図のステップS21では、上述の選択テーブルから出力書体の選択データを読み出す。この選択データの読み込みは、書体が変更された場合にのみ行なうようにしても良い。また、続くステップS22では、文字コードより、文字に1対1に対応したエレメント組み合わせ情報をフォントファイルより読み出す。

【0023】図11は、このエレメント組み合わせ情報のデータを表わしたもので、要素として、ハイブリッドフォントの生成が可能かどうかを判断するフラグ（以後、ハイブリッドフラグという）、次に、その文字を構成する各エレメント識別番号とそのエレメントに対しビットマップデータの有無を判別するフラグ（以後、ビットマップフラグという）、及び、各エレメントをセットする座標を示すエレメント配置情報（位置情報データ）からなる。

【0024】ステップS23では、輪郭発生を一意に切り替える処理を行ない、例えば、PRT12やCRT7等の出力デバイスが品位を要求しない場合、切り替えフラグを自動または手動によりONとして、処理をステップS27に進める。また、ステップS24では、選択データのハイブリッド閾値と出力文字サイズとを比較し、文字サイズが閾値より大きい場合、処理をステップS27に進める。さらに、ステップS25では、ハイブリッドフラグを判定する部分で、フラグがONならステップS26に、フラグがOFFならば、処理をステップS27に進める。

【0025】なお、ステップS26では、後述するハイブリッドフォントを発生させ、また、ステップS27では、アウトラインフォントを発生させる。図10のステップS26は、図5のステップS3に、ステップS27は、図5のステップS4での処理にそれぞれ対応する。図12は、図10のステップS26（図5のステップS3）に対応するハイブリッドフォントの発生処理手順を示すフローチャートである。同図のステップS31では、上述のレイトテーブルより、出力文字サイズからレイト及び使用するビットマップサイズを読み出す。ステップS32では、先に読み出したエレメント組み合わせ情報より、順次、エレメントID及びビットマップフラグを読み出し、フラグON（ビットマップ有り）ならば、処理をステップS33に進め、フラグがOFFなら

ば、ステップ S 3 5 に行く。

【0026】ステップ S 3 3 では、エレメント ID、及びステップ S 3 1 で読み出した使用ビットマップサイズよりフォントデータを検索し、それが見つければステップ S 3 4 へ、検索ができなければステップ S 3 5 へそれぞれ処理を進める。このステップ S 3 5 では、エレメント ID よりアウトラインストロークデータを読み出し、出力文字サイズに合ったビットマップストロークを生成する。また、ステップ S 3 4 では、検索したビットマップデータを読み出し、続く、ステップ S 3 6 で、ステップ S 3 1 で読み出したレイトを用いて、ビットマップデータのサイズ変換を行なう。

【0027】ステップ S 3 7 では、生成されたビットマップデータを配置情報をもとに合成する。そして、ステップ S 3 8 で、文字を構成するエレメントデータが全て合成されたかを判断し、合成終了ならば本処理を終了し、合成が未終了ならば、処理をステップ S 3 2 へ戻す。図 1 3 は、図 1 0 のステップ S 2 7 (図 5 のステップ S 4) に対応するアウトラインフォント発生処理手順を示すフローチャートである。同図のステップ S 4 1 では、先に読み出したエレメント組み合わせ情報より、順次、エレメント ID を用いてアウトラインエレメントデータを読み出す。続くステップ S 4 2 では、読み出したデータを用いて、出力文字サイズに合ったビットマップエレメントを生成する。そして、ステップ S 4 3 では、生成されたビットマップデータを、配置情報をもとに合成する。

【0028】ステップ S 4 4 では、文字を構成するエレメントデータが全て合成されたかどうかを判断し、合成途中ならば、処理をステップ S 4 1 へ戻し、また、合成終了と判断されたならば、本処理を終了する。本実施例に係るシステムにおける基本動作は、上述の処理を経て、図 5 のステップ S 5 での処理、つまり、合成したビットマップデータを出力用バッファにセットし、CRT 7 や PRT 1 2 に所定の文字出力を行なう。

【0029】以上説明したように、本実施例によれば、複数の文字エレメントをベクトル形式で記憶する場合、小さな文字を出力する際に、文字の潰れが発生するエレメントに対してのみビットマップで記憶し、他のエレメントについてはアウトラインにより発生させ、合成することにより、装置に搭載するビットマップデータサイズを大幅に減らし、かつ、十分な文字品位を保つことが可能となる。

【第 2 実施例】次に、本発明の第 2 の実施例について説明する。

【0030】ここで扱うのは、例えば、20 ドット以下の比較的小さな出力文字サイズで文字を出力する場合である。そして、一般にこのような出力サイズでは、フォントを簡略化して出力することが行なわれている。本実施例では、ハイブリッドフォントのビットマップ部分を

取り去り、文字を簡略化して表示する方法(以後、これをシンプルフォントという)について説明する。

【0031】図 1 4 は、本実施例に係るシステムにおける基本動作を示すフローチャートである。なお、同図において、ステップ S 5 1 は、図 5 のステップ S 1 に対応し、同様に、ステップ S 5 2 は、図 5 のステップ S 2 に、ステップ S 5 4 は、図 5 のステップ S 3 に、そして、ステップ S 5 5 は、図 5 のステップ S 4 にそれぞれ対応する。

【0032】そこで、本実施例では、図 5 に示す、上記第 1 実施例に係る処理と相違のあるデータ構造、及びその他の処理手順について説明する。まず、データ構造の相違として、図 8 に示す輪郭発生選択データの構造に対して、ハイブリッドフォント閾値の下限を設定するパラメータを追加する。言うまでもなく、図 9 に示すレイトテーブルの使用ビットマップサイズにシンプルフォントフラグを設定し、そのフラグにより、シンプルフォントの採用を決定してもよい。

【0033】図 1 4 のステップ S 5 2 では、入力された書体、文字コード、出力文字サイズ、輪郭発生手段切り替えフラグ(上記切り替えフラグと同じ)により、輪郭発生手段をハイブリッドフォント、アウトラインフォント、あるいはシンプルフォントによる生成か否かを選択する。図 1 5 は、輪郭発生手段の生成過程を示すフローチャートである。同図のステップ S 6 1 では、上記第 1 実施例と同様、選択テーブルから出力書体の選択データを読み出す。なお、この選択データの読み込みは、書体が変更された場合にのみ行なうようにしても良い。

【0034】ステップ S 6 2 では、文字コードより文字に 1 対 1 に対応したエレメント組み合わせ情報をフォントファイルより読み出す。続くステップ S 6 3 では、輪郭発生を一意に切り替える処理を行ない、例えば、出力デバイスにより手動及び自動的に切り替えフラグを切り替え、その切り替えに従って、処理をステップ S 5 3、S 5 4、あるいはステップ S 5 5 に進める。

【0035】ステップ S 6 4 では、選択データのハイブリッド閾値と出力文字サイズとを比較し、文字サイズが閾値より大きい場合、処理をステップ S 5 5 に進める。また、ステップ S 6 5 では、選択データのシンプルフォント閾値と出力文字サイズとを比較し、文字サイズが閾値より小さい場合、処理をステップ S 5 3 に進める。ステップ S 6 6 では、ハイブリッドフラグを判定する処理を行ない、ここで、フラグが ON と判定されたならば、ステップ S 5 4 に、また、フラグが OFF ならば、処理をステップ S 5 5 に進める。

【0036】そこで、ステップ S 5 3 でのシンプルフォント発生処理について説明する。シンプルフォントとは、基本的にハイブリッドフォントのビットマップ部分を取り去り、アウトラインにより生成されるエレメントのみで文字を生成することを指す。しかし、ビットマッ

ブ部分以外にも冗長となる部分もあるため、エレメント組み合わせ情報にビットマップフラグと同じ要領でシンプルフォントフラグを付加し、そのフラグのON/OFFにより、エレメントの表示を制御するようにしても良い。言うまでもなく、エレメント部分をその芯線で表示しても良い。

【0037】図16は、シンプルフォント発生処理の過程を示すフローチャートである。同図のステップS71では、先に読み出したエレメント組み合わせ情報より、順次、エレメントID及びビットマップフラグを読み出し、フラグがON（ビットマップ有り）と判定されたならば、処理をステップS74に進め、フラグがOFFならば、ステップS72の処理に入る。すなわち、ステップS72では、アウトラインエレメントのビットマップ化を行なう。

【0038】ステップS73では、生成されたビットマップデータを、配置情報をもとに合成する。そして、ステップS74では、文字を構成するストロークデータが全て合成されたかどうかを判断し、合成途中ならば、処理をステップS71へ戻し、合成終了ならば、本処理を終えて、図14のステップS56へ行く。図14のステップS56では、合成したビットマップデータを出力用バッファにセットし、PRT12やCRT7に文字出力を行なう。

【0039】図17は、図2に示す文字エレメントと同様、実線部が基本エレメントであり、破線部が基本エレメントを修飾するエレメントである。そして、図17に示す文字に対して、本実施例に係るシンプルフォントを適用してビットマップ部分を取り去ると、図18に示す表示が得られる。このように、本実施例によれば、文字品位を要求しない文字についてはビットマップ部分を不表示として、文字を簡略して表示することにより、文字出力の高速化が可能となる。

【0040】本発明は、複数の機器から構成されるシステムに適用しても1つの機器から成る装置に適用しても良い。また、本発明は、システムあるいは装置にプログラムを供給することによって達成される場合にも適用できることは言うまでもない。

【0041】

【発明の効果】以上説明したように、本発明によれば、文字データの出力サイズに応じて文字輪郭の発生種別を切り替えることで、装置に搭載するビットマップデータサイズを大幅に減らし、かつ、十分な文字品位を保つことが可能となる。また、他の発明によれば、文字品位を要求しない文字についてはビットマップ部分を不表示とすることにより、文字出力の高速化が可能となる。

【図面の簡単な説明】

【図1】本発明の実施例に係るシステムの構成を示すブロック図である。

【図2】文字エレメントの種類を表わす図である。

【図3】アウトラインエレメントのデータを規定する図である。

【図4】アウトラインエレメントのデータを規定する図である。

【図5】実施例に係るシステムでの基本動作を示すフローチャートである。

【図6】輪郭発生選択テーブルの作成過程を示すフローチャートである。

【図7】実施例に係る基本的なフォントファイルの構成を示す図である。

【図8】輪郭発生選択データの構造を示す図である。

【図9】レイアウトテーブルの構成例を示す図である。

【図10】輪郭発生選択の動作手順を示すフローチャートである。

【図11】エレメント組み合わせ情報を示す図である。

【図12】ハイブリッドフォントの発生処理手順を示すフローチャートである。

【図13】アウトラインフォントの発生動作を示すフローチャートである。

【図14】本発明の第2の実施例に係る基本動作を示すフローチャートである。

【図15】輪郭発生選択の動作手順を示すフローチャートである。

【図16】シンプルフォントの発生動作を示すフローチャートである。

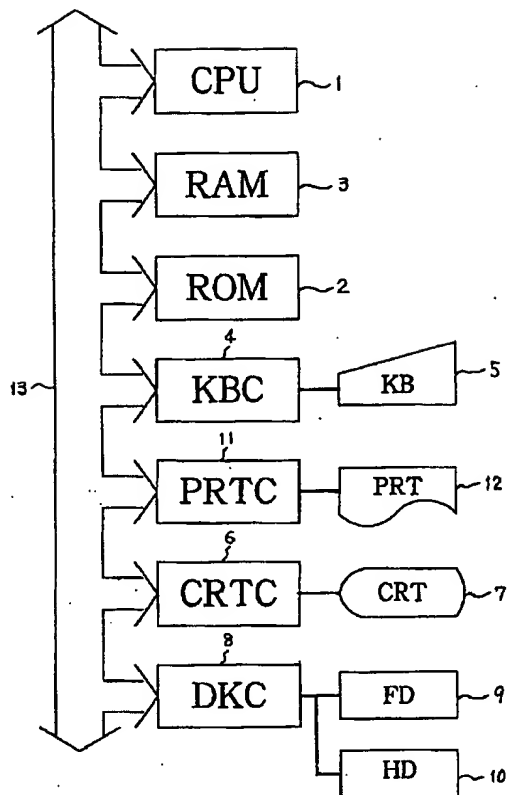
【図17】シンプルフォント発生概念を示す図である。

【図18】シンプルフォント発生概念を示す図である。

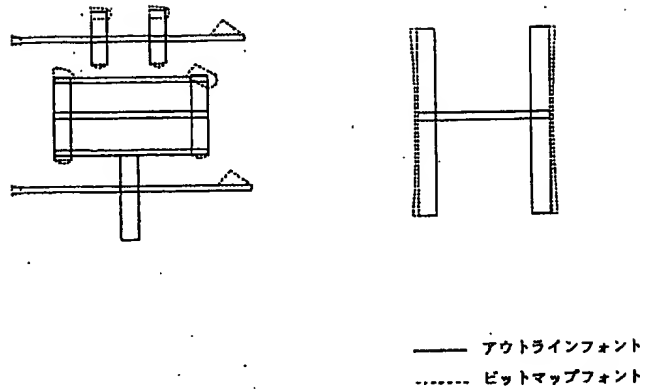
【符号の説明】

- 1 CPU（中央処理装置）
- 2 ROM（読み出し専用メモリ）
- 3 RAM（ランダムアクセスメモリ）
- 4 KBC（キーボード制御部）
- 5 KB（キーボード）
- 6 CRTC（ディスプレイ制御部）
- 7 CRT（ディスプレイ装置）
- 8 DKC（ディスク制御部）
- 9 FD（フロッピーディスク装置）
- 10 HD（ハードディスク装置）
- 11 PRTC（プリンタ制御部）
- 12 PRT（プリンタ装置）
- 13 システムバス

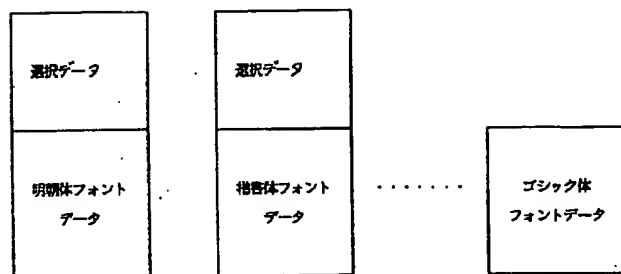
【図1】



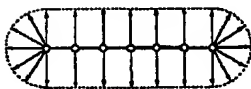
【図2】



【図7】



【図3】



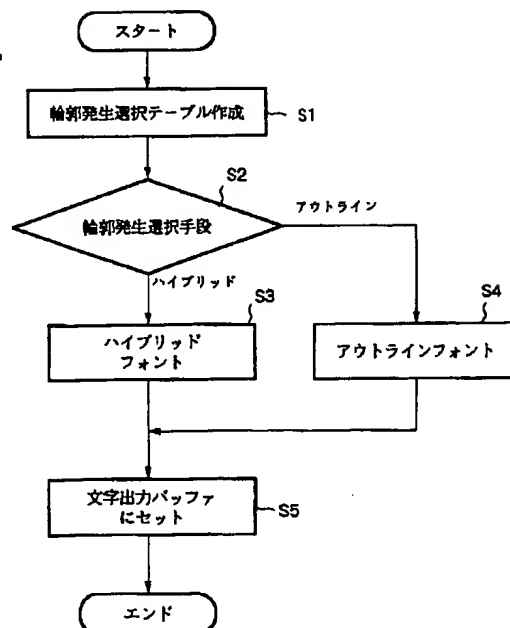
【図4】



【図9】

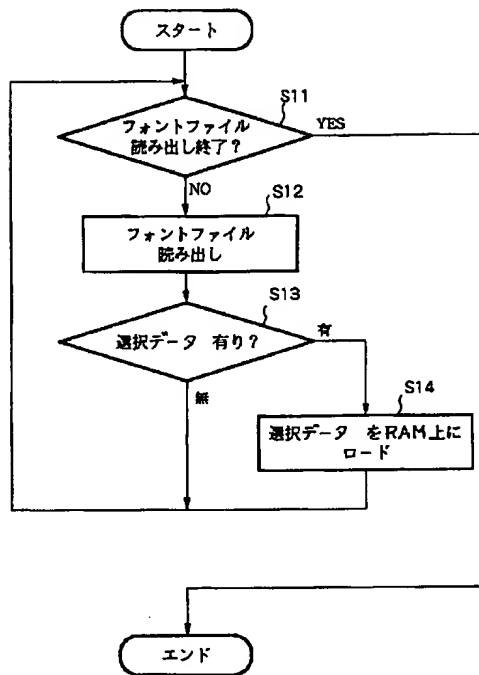
出力ビット数	拡大・縮小率 (rate)	使用ビット数
10	4/5	12
11	8/7	12
12	0/0	12
.	.	.
.	.	.
58	24/25	60
59	30/31	60
60	0/0	60

【図5】





【図 6】

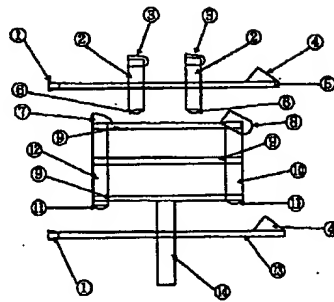
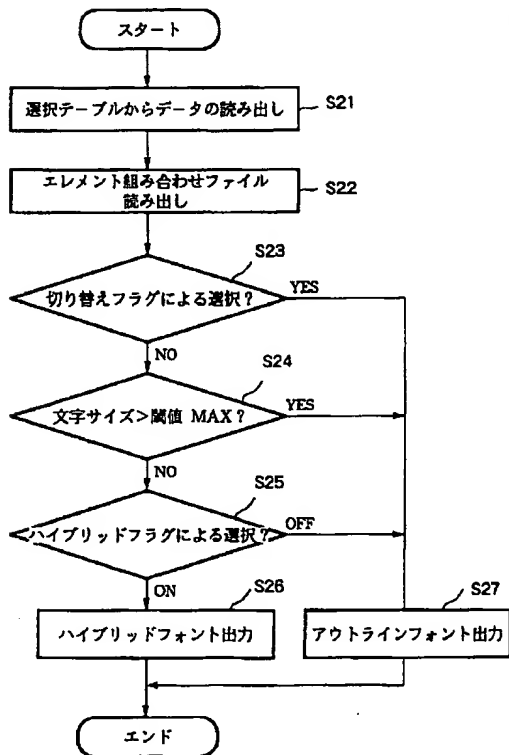


【図 8】

書体名
ビットマップ開始 (MIN)
ビットマップ終了 (MAX)
ビットマップ数
ビットマップサイズテーブル
18
20
60
レイトテーブル

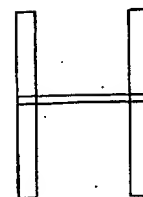
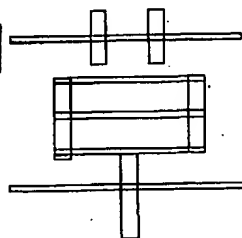
【図 11】

【図 10】

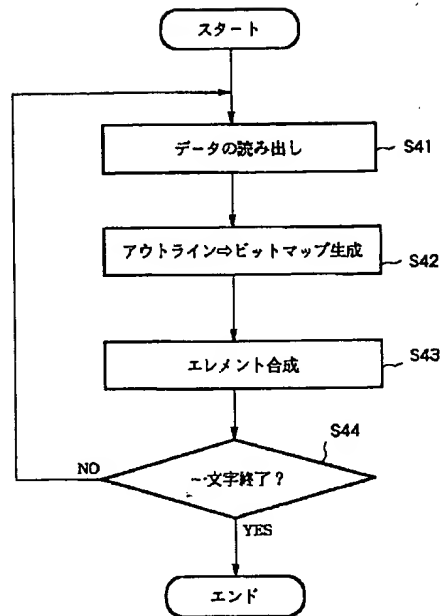


ハイブリッドフォントフラグ	
ストロークID	ビットマップ
①	なし
②	なし
③	あり
④	なし
⑤	あり
⑥	あり
⑦	あり
⑧	あり
⑨	なし
⑩	なし
⑪	あり
⑫	なし
位置情報データ	

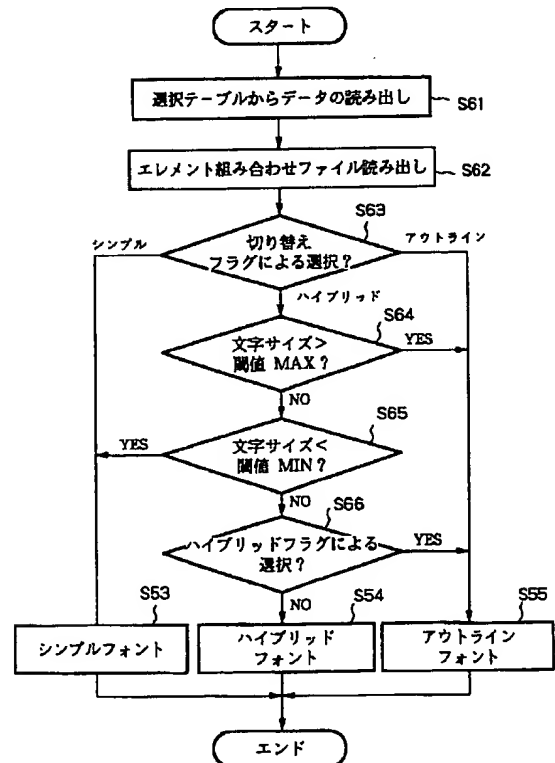
【図 18】



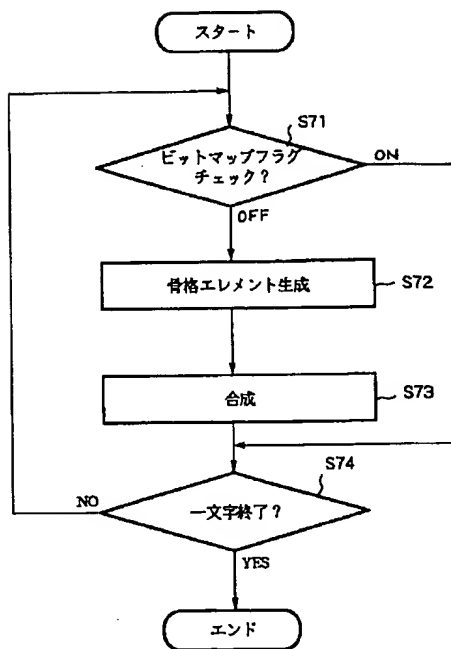
【図 13】



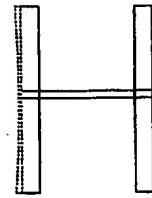
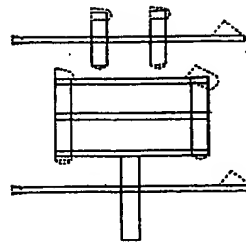
【図 14】



【図 16】



【図 17】



—— アウトラインフォント  
..... ビットマップフォント